APPLICATION UNDER UNITED STATES PATENT LAWS

Atty. Dkt. No. __PM 274072/A0000133_____
<div align="center">(M#)</div>

Invention: A METHOD OF ANALYZING GENE EXPRESSION DATA USING FUZZY LOGIC

Inventor (s):   WOOLF, Peter J.
               WANG, Yixin

Pillsbury Winthrop LLP
Intellectual Property Group
1100 New York Avenue, NW
Ninth Floor
Washington, DC 20005-3918
<div align="center">Attorneys</div>
Telephone: (202) 861-3000

<u>This is a:</u>

☐ Provisional Application

☒ Regular Utility Application

☐ Continuing Application
   ☒ The contents of the parent are incorporated
     by reference

☐ PCT National Phase Application

☐ Design Application

☐ Reissue Application

☐ Plant Application

☐ Substitute Specification
   <u>Sub. Spec</u> Filed _____
       in App. No. _____/_____

☐ <u>Marked up Specification re</u>
   Sub. Spec. filed _____
       In App. No _____/_____

SPECIFICATION

A METHOD OF ANALYZING GENE EXPRESSION DATA USING FUZZY LOGIC

RESERVATION OF COPYRIGHT

BACKGROUND OF THE INVENTION

1. Field of the Invention

10   The invention relates to a data processing system and method of use for

analyzing gene expression data using a fuzzy logic-based computer algorithm.

2. Description of Background Information

Cells regulate the expression of their genes in response to environmental

15   changes. Normally this regulation is beneficial to the cell, protecting it from

starvation or injury; however errors in this regulation can lead to serious diseases

ranging from cancer to heart disease. Measuring the differential expression of genes

from various stages of an organism's development, different tissues, and organisms

subjected to different stresses provides information instrumental in understanding the

20   relationships between genes and their functions. Gene regulation is useful for both

assaying drugs and as a source of new molecular targets, assuming the regulatory

network controlling a given gene is well understood. As such, changes in gene

expression patterns can be used to assay drug efficacy throughout the drug discovery

process.

One assay that takes advantage of the existing level of sequence information, and that is complementary to sequence and genetic analysis, is gene expression profiling. Expression profiling can be carried out by one of a number of different technologies, such as commercially or privately manufactured gene chips, which

5   typically measure the expression level of thousands of genes simultaneously using an array of oligonucleotides bound to a silicon surface. These arrays are hybridized under stringent conditions with a complex sample representing mRNAs expressed in the test cell or tissue. Target sequences hybridize to immobilized oligonucleotides and are typically detected via fluroescent labels. Relative intensity levels of

10   fluorescent labels indicate relative gene expression in a given sample obtained from a source subjected to a particular condition. As a sample source is subjected to a variety of conditions, a given gene will display a profile under these conditions. Thus, gene profiles represent a vector of gene expression intensities corresponding to multi-conditional gene expression patterns. The results from these expression profiling

15   technologies are quantitative and highly parallel, thereby allowing an accurate snapshot to be made of the workings of the cell in a particular state.

Since thousands of hybridization reactions may occur in a single array, expression profiling assays generate huge data sets that are not amenable to simple analysis. To maximize the use of such data, efforts are underway to develop

20   algorithms interpreting and interconnecting results for different genes under different conditions. Currently, expression data is typically analyzed using clustering techniques comprising algorithms that identify distinct expression patterns by grouping genes with similar expression patterns (Tavazoie,1999; Cho, 1998). An example of such a clustering algorithm is CAST (Cluster Affinity Search Technique),

25   which uses a graph theoretic approach and employs a stochastic model of input

2

(Yakhini, 1998). Cluster analysis of multi-conditional gene expression patterns generally involve the steps of i) measuring gene expression levels reported as a vector of real numbers; ii) computing a similarity matrix for the measured genes; iii) clustering genes based on their similarity to each other; iv) visually representing the

5   clusters; and v) analyzing the results obtained therefrom.

Clustering, however, can only distinguish between those genes that have the same and different expression profiles. Genes in any given cell make up a complex network that cannot be revealed with current techniques such as clustering, etc. In order to determine networks describing how various genes interrelate, more elaborate

10   data mining techniques are needed.

Fuzzy logic uses techniques drawn from engineering and other applied sciences for controlling systems as diverse as washing machines to auto-focus cameras (Cox, 1992; Zadeh, 1974). The concept of fuzzy logic is based on the "fuzzy estimation" of human thinking, as opposed to precise mathematical computation.

15   Fuzzy logic provides a means for transforming precise numbers, such as 12.433, into qualitative descriptors, such as HIGH in a process called "fuzzification". Thus, discrete input values may be converted into a range of values. Once transformed, such qualitative data may be analyzed using heuristic rules, which in turn generate fuzzy solutions. For example, the heuristic rule "if HOT then move FAST" takes HOT as a

20   fuzzy input and FAST as a fuzzy solution. In another process called "defuzzification", this heuristic solution can be transformed from a qualitative descriptor back into a precise number.

3. Definitions

For purposes of clarification, and to facilitate an understanding of the present invention and the embodiments disclosed herein, a number of terms used herein are defined as follows:

5

Expression profiling:

A process by which molecular techniques are used to measure and compare expression levels of certain nucleic acid sequences (e.g., mRNAs, genes, expressed sequence tags (ESTs)), or levels of certain gene products, such as amino acid sequences (protein or fragments of proteins), in a cell-derived sample in relation to the levels of the same nucleic acid sequences or gene products from a different sample, or from the same sample measured at a different time point.

Gene:

A sequence of nucleotides specifying a particular polypeptide chain.

Gene product:

One or several polypeptide chains of amino acids translated from RNA transcribed from a gene.

Activator/Repressor:

Proteins, such as transcription factors and tumor suppressors, capable of inducing physiological change, usually enhancement or inhibition of gene expression in a given biological system (activation or deactivation), typically in response to an environmental stress.

mRNA:

Messenger Ribonucleic acid.

25

4

SUMMARY OF THE INVENTION

An embodiment of the invention provides a method for managing and analyzing information obtained from differential expression of genetic information in biological cells. Crisp input data is received from sets of expression data from control

5 and treatment cell-derived samples representing a direction and a magnitude of regulation of each one of a higher number of different genes or proteins. The crisp input data is fuzzified to provide fuzzified values. A set of heuristic rules is applied to the fuzzified values to generate a predicted value of a data point C. The predicted value of the data point C is defuzzified. Finally, a confidence level of the predicted

10 value of C is determined.

A second embodiment of the invention provides a system for managing and analyzing information obtained from differential expression of genetic information in biological cells. The system includes a data receiver for receiving crisp input data from control and treatment cell of cell-derived samples. A fuzzifier fuzzifies the crisp

15 input data to provide fuzzified values. A heuristic rules applier applies a set of heuristic rules to the fuzzified values to generate a predicted value of a data point C. A defuzzifier defuzzifies the predicted value of C. Finally, a confidence level determiner determines a confidence level of the predicted value of C.

A third embodiment of the invention provides a machine-readable medium

20 having recorded thereon machine-readable information. When the machine-readable information is loaded into a computer memory and executed by a computer, the machine-readable information causes the computer to: receive crisp input data from sets of expression data from control and treatment sets of cell-derived samples representing a direction and a magnitude of regulation of each one of a higher number

25 of different genes or proteins; fuzzify the crisp input data to provide fuzzified values;

5

apply a set of heuristic rules to the fuzzified values to generate a predicted value of a

data point C; defuzzify the predicted value of C; and determine a confidence level of

the predicted value of C.

BRIEF DESCRIPTION OF THE DRAWINGS

5   Illustrative embodiments of the invention are further illustrated in the

accompanying drawings in which like reference numerals represent the same or

similar elements throughout the several views of the drawings.

Figure 1 depicts fuzzy membership as a function of a normalized expression

level.

10   Figure 2 shows a decision matrix describing an activator (A) and repressor (B)

acting on a target (C).

Figures 3A-3C show the fuzzy logic prediction for CYB2 (A), HAP1 (B) and

CYC1 (C).

Figure 4 shows the HAP1 and ROX1 regulatory network as predicted by the

15   yeast expression data set.

Figure 5 is a block diagram of an expression profiling data analysis system.

Figure 6 is a block diagram which illustrates the analyzer in detail.

Figure 7 provides a more detailed view of the confidence level determiner of

Figure 6.

20   Figure 8 provides a more detailed view of the filter of Figure 6.

Figure 9 provides a more detailed view of the scorer of Figure 6.

Figure 10 is a flow diagram of a gene expression profiling process.

Figure 11 is a flow diagram of an embodiment of the analyzer system.

DETAILED DESCRIPTION

Three main advantages favor the application of fuzzy logic to the analysis of gene expression data over previously known techniques. First, fuzzy logic inherently accounts for noise in data because the algorithm extracts trends, not precise values.

5   Second, in contrast to other automated decision making algorithms, such as neural networks or polynomial fits, algorithms in fuzzy logic are cast in the same language used in day-to-day conversation. As a result, predictions made using fuzzy logic are easily interpreted and can be extrapolated in predictable ways. Third, fuzzy logic techniques are computationally efficient and can be scaled to include an unlimited

10   number of components. Thus, they are able to recognize a large number of biologically important patterns.

One embodiment of the invention comprises a data processing system, comprising an analysis system 10, as shown in Fig. 5. An analyzer 27, included within host computer 24, is used for carrying out certain analysis processes associated

15   with expression profiling and managing data acquired from the expression profiling . Analyzer 27 comprises fuzzy logic that can identify logical relationships between genes, and in some cases, even predict the function of an unknown gene.

This algorithm was validated using yeast expression data gathered from the Affymetrix GeneChip® system. By using yeast gene expression data collected at

20   different time points of the cell cycle, the illustrated analyzer 27 permits identification of many regulatory elements and their target genes within the yeast cell that work together to maintain and control the "state" of the cell. Several cases were validated by available experimental results, including the signaling network controlled by the transcription factors HAP1 and ROX1, known to control the transition in yeast from

25   anaerobic to aerobic growth. These results suggest that the fuzzy logic technique of

7

the illustrated embodiment can effectively find biologically relevant connections between sets of genes, which in turn aids in describing the complex web of interactions that regulate gene expression.

Referring now to the drawings in greater detail, Figure 5 shows an analysis system 10 according to the illustrated embodiment of the present invention. An expression profiling subsystem 12 is provided, which is coupled to a client computer 14. In the illustrated embodiment the expression profiling subsystem 12 is coupled via intranet 22; however, other methods, such as, but not limited to, direct coupling may also be employed. Client computer 14 comprises, among other elements, a browser application 16, a human interface 18, and a display 20. Human interface 18 may comprise any standard or other interface for facilitating human interaction with and control of client computer 14, including, for example, a keyboard and a mouse. Client computer 14 is coupled to a host computer 24 via, for example, a network connection illustrated in Fig. 5 as the intranet 22. Host computer 24 is connected to a database 26.

Expression profiling system 12 may comprise, for example, an Affymetrix cDNA array. It generates, from control and treatment sets of cell-derived samples, respective sets of sequence data representing a direction and a magnitude of regulation of each one of a high number of different nucleic acid sequences.

Client computer 14, together with human interface 18, display 20, and browser application 16, allow a user to operate analysis system 10. Client computer 14 communicates with database 26 through intranet 22 and host computer 24. Expression profiling subsystem 12 obtains the expression profiling data and stores that data in an organized fashion on database 26.

Host computer 24 is provided with, among other elements, the analyzer 27 for carrying out certain analysis process steps associated with expression profiling and managing the data acquired from the expression profiling. A database server software component 28 is provided for handling and acting on database queries and responses.

5 Figure 6 shows the analyzer 27 in more detail. The analyzer receives crisp input data, from sets of expression data, into data receiver 60. Optionally, the data is filtered by filter 62. The filtered data is then fuzzified, by fuzzifier 64, into fuzzy values. In an embodiment of the invention, the fuzzy values have high, medium and low components. The fuzzy values are applied, by heuristic rules applier 66, to a

10 decision matrix. The heuristic rules applier 66 provides a fuzzy value for a predicted value of a data point C. Defuzzifier 68 defuzzifies the fuzzy value of C. Confidence level determiner 70 determines a level of confidence in the predicted value of C.

Figure 7 is a more detailed view of the confidence level determiner 70. The confidence level determiner includes a calculator 702 for calculating a difference r

15 between the defuzzified predicted value of a data point C and an observed value of C. Squarer 704 squares r to provide r^2. Scorer 708 includes a multiplier 7082 (see Figure 9) for multiplying a distribution variance by r^2 to provide a general score for predicting credibility of the predicted value of C.

Figure 8 shows a more detailed view of filter 62. The filter 62 includes an

20 accepter 622. The accepter 622 accepts crisp input data only when one of the crisp input data, having a value greater than all other ones of the crisp input data, is at least three times larger than another one of the crisp input data having a value less than all other ones of the crisp input data.

The heuristic rules applier 66 applies heuristic rules, as illustrated by Figure 2,

25 which shows the heuristic rules as applied to a decision matrix. The heuristic rules of

9

Figure 2 illustrate, for example, an activator/repressor model, although other models

may also be used depending upon the type of relationship under study. Assuming A is

an activator, B is a repressor and C is a target, the matrix of Figure 2 shows the

expected relationships. For example, when A is HI and B is HI, C is MED. When A

5   is HI and B is MED, C is HI. When A is HI and C is LO, C is HI. When A is MED

and B is HI, C is LO. When A is MED and B is MED, C is MED. When A is MED

and B is LO, C is HI. When A is LO and B is HI, C is LO. When A is LO and B is

MED, C is LO. Finally, when A is LO and B is LO, C is MED.

Figure 10 generally shows an expression profiling process in accordance with

10   the illustrated embodiment. In an initial act S2, gene expression values are generated

based upon a baseline sample (otherwise referred to as a control sample) of cells. In

an act S6, one or more gene expression values may be generated based upon treated

samples, i.e., samples of cells based upon those cells entering into a diseased state or

being treated with a particular compound. After performing each of acts S2 and S6,

15   quantitation is performed.

In order to determine whether gene expression levels have been substantially

affected (i.e., either repressed or induced), the expression level of the genes generated

in a baseline sample is compared with the expression level of the genes generated and

grouped in the treated sample or samples, to produce, for each treated sample, an

20   indication of whether a gene was regulated and the extent and direction of that

regulation.

More specifically, by way of example, a sample of RNA may be monitored

using an expression profiling array, such as an Affymetrix GeneChip™ probe array

having, for example, oligonucleotides corresponding to the human genome, capable of

25   detecting expression levels for over 6,000 sequences for that genome. Affymetrix

10

provides a GeneChip™ fluidics station that automates the hybridization of nucleic acid targets to a probe array cartridge, and thus controls the delivery of reagents and the timing and temperature for hybridization. Each fluidics station can independently process four probe arrays at a given time.

5   Accordingly, each target may be prepared from a set of cell dishes by isolation of RNA over a course of time. The treatment of those cells may be emulated by adding, for example, serum thereto. At predetermined intervals, a small amount of the fluid is removed, and the cells are put in a quiescent state to stop the reaction time. Accordingly, a large set of targets, having a predetermined amount of liquid (e.g., .5

10   ml each) is produced. The GeneChip™ fluidics station will then automatically hybridize each target, i.e., it will extract all the RNA and label the RNA by adding a chemical tag to each molecule, and control the delivery of the resulting liquid to the probe arrays to facilitate obtaining sequencing information regarding the mRNAs. This is done by the probe arrays exposing the target to light at a predetermined

15   location and measuring the photons collected at various locations within the arrays. The amount of mRNA is then ascertained based upon the signal strength of the reading given by the probe at the appropriate location corresponding to that sequence or sequence segment. A net change in signal may indicate activation or repression of gene transcription, or possibly post-transcriptional stabilization the mRNA due a

20   given set of treatment conditions.

The expression data so obtained are then filtered to create suitable sets of data for analysis according to the invention, essentially as diagrammed in Figure 11. Such filtering serves to ensure selected data are above a minimum noise threshold and represent gene whose net change in expression satisfies a predetermined level of

25   signal strength.

The fuzzy logic algorithm may also be used to analyze data sets generated by translated products of genes expressed in biological cells under control and treatment conditions. For example, the ProteinChip™ System (Ciphergen Biosystems, Palo Alto, CA) consists of microchips having activated surfaces that allow immobilization

5   of antibodies, receptors or DNA for capturing proteins from a sample. After sample binding, the array is washed to remove unbound molecules. Thereafter, energy adsorbing molecules are added to facilitate detection captured proteins via, for example, mass spectrometry. Multiple sets of protein expression spectra may be obtained and filtered to create suitable sets of data for analysis according to the

10   illustrated embodiments of the invention.

Certain implementations or embodiments of the invention are further illustrated in the following, nonlimiting example.

EXAMPLE

Public domain GeneChip® data describing the yeast cell cycle (downloaded

15   from http://genomics.stanford.edu) was chosen to validate the fuzzy logic algorithm. Since the yeast cell cycle is a tightly regulated process at the genetic level, the expression data was expected to show detectable relationships among different genes that might be detected via a fuzzy logic algorithm. Also, years of experimental work on yeast has generated an extensive body of biological data describing many of the

20   proteins in the organism, allowing the confirmation or dismissal of findings based on data in the literature.

In this illustrated embodiment, at S20, GeneChip data were obtained from a set of 17 experiments in which data points were profiled under various biological conditions. Thus, the relationship between proteins in numerous distinct cell types

25   and under various stress conditions were followed to ensure detected relationships

12

between gene products were not the results of an artifact created under a particular experimental condition. Each individual experimental condition was repeated 2 or 3 times, and the averaged results were included as a single experiment in the set of 17 experiments analyzed. All proteins later evaluated for relationship purposes must

5   have been expressed to some measurable degree under all 17 experimental conditions. Preferably, a minimum of 9 experimental conditions should be included, although this is not absolute. No maximum number of experiments is controlling.

Before expression data were analyzed, the data were filtered, at S22, to ensure that 1) the data are above a threshold noise level that is determined by GeneChip®

10   software and 2) the data set includes genes that significantly differ in their level of expression. For the yeast data set, the noise level was assumed to be 30 in average difference; thus the highest member of a particular data set had to exceed the noise level to allow that set into the calculation. In this embodiment, for a set to be selected, its regulation magnitudes must vary over a certain minimum range, so as to ensure

15   that the observed signal change was statistically significant. In the illustrated embodiment, the maximum value in the set was at least 3 times greater than the minimum value. The threshold difference between maximum and minimum gene expression levels depends upon the sensitivity of the detection system collecting the data, and thus may vary according to the means of detection selected. The fuzzy logic

20   algorithm of this example processed only those genes that met both criteria.

In this example, genes were selected that follow the pattern of a gene product (C) controlled by both an activator (A) and repressor (B), although any pattern can be searched for in a pre-selected pattern. Generally, the activator-repressor model provides that the concentration of the target C will be high when the activator is high

25   and the repressor is low. Conversely, when the repressor concentration is high, and

the activator is low, the concentration of the target is low. An unlimited number of models may be employed, and thus the invention is not limited to the interactions within triplet subsets as exemplified.

In order to analyze the genetic expression data, the data are transformed, at

5   S24, from crisp values to fuzzy values in a process called "fuzzification." Data are fuzzified by first normalizing the data from 0 to 1. Thereafter, the normalized value is broken up into various membership classes. For example, Figure 1 shows the three fuzzy sets used in this algorithm, HI, MED, and LO, as a function of the normalized value. Note that three fuzzy sets are used as an example only and that the number of

10   fuzzy sets need not be three, for example, five, or any other appropriate number may be used. For a normalized value of 0.25, the fuzzy value is 0.5 LO, 0.5 MED, and 0 HI; or put another way, 0.25 is 50% low, 50% medium and 0% high. Accordingly, any crisp, normalized value can be broken down into its membership in each fuzzy class.

15   After the data are fuzzified, triplets of data are compared, at S26, using a set of heuristic rules in the form of a decision matrix (see Figure 2). Fuzzified values of A and B are entered into this matrix, and at points where their predictions overlap a score is generated. The goal of this process is to identify values of A and B that overlap and to give that element its appropriate weight. The following pseudocode

20   may be used to apply the fuzzy values to a decision matrix.

The matrix is as depicted, wherein "hm" means A is high and B is med.

	B high	B med	B low
A high	hh	hm	hl
A med	mh	mm	ml
A low	lh	lm	ll

25

14

Data enter the matrix in the form of fuzzified values, called fuzzA and fuzzB.

```
5       /* zero out matrix */
        hh=hm=hl=mh=mm=ml=lh=lm=ll=0;
        /* zero out count*/
        count=0;

10      /* look for overlap, and if overlap add to the count */

        /*high A and high B*/
        if((fuzzA[HI]*fuzzB[HI] != 0.0)
        {
15              hh=fuzzA[HI]+fuzzB[HI];
                count++;
        }

        /*high A and med B*/
20      if((fuzzA[HI]*fuzzB[MED] != 0.0)
        {
                hm=fuzzA[HI]+fuzzB[MED];
                count++;
        }

25

        /*repeat for all combinations of A and B (total of 9 comparisons)*/

        /*normalize scores based on count such that each condition gets equal
        weight */
30      /* Note that the only possible values of count are 4, 3, 2 and 1 */

        if(count==4)
        {
                hh=hh/4;
35              hm=hm/4;
                hl=hl/4:
        mh=mh/4;
                mm=mm/4;
                ml=ml/4:
40      lh=lh/4;
                lm=lm/4;
                ll=ll/4:
        }
        if(count==2)
45      {
                hh=hh/3;
                hm=hm/3;
                hl=hl/3:
        mh=mh/3;
50              mm=mm/3;
                ml=ml/3:
        lh=lh/3;
                lm=lm/3;
                ll=ll/3:
55      }
        if(count==1)
        {
                hh=hh/2;
                hm=hm/2;
```

```
            hl=hl/2:
    mh=mh/2;
            mm=mm/2;
            ml=ml/2:
5   lh=lh/2;
            lm=lm/2;
            ll=ll/2:
    }
```

```
10  /*this process can be repeated for all of the experiments to get average
    values for hh, hm, hl, etc*/
```

15   The following pseudocode illustrates S28, predicting the value of a data point
C and defuzzifying the value/

```
20  /* we need to defuzzify values using, for example, the activator/repressor schema, as shown in the
    heuristics table of Figure 2 (or whatever kind of relation we are looking for), which is stored in
    TempTable[2][2].  TempTable[2][2] is a 3x3 array having elements ranging from TempTable [0][0]
    through TempTable [2][2] */

25  /* If heuristics table shows that a HI result is expected for C, when A is HI and B is HI,
    then ... */
    if(TempTable[A_HI][B_HI]==HI) /*schema at A_HI,B_HI is HI*/
    {
                    Cval=Cval+hh;  /* Note hh is effectively being multiplied
30                                      by HI value 1 */
    }

    /* If heuristics table shows that a MED result is expected for C, when A is HI and B is HI,
    then ... */
35  if(TempTable[A_HI][B_HI]==MD) /*schema at A_HI,B_HI is MED*/
    {
                    Cval=Cval+hh/2; /* Note hh is effectively being multiplied
                                        by MED value 0.5 */
    }
40
    /* If heuristics table shows that a LO result is expected for C, when A is HI and B is HI,
    then ... */
    if(TempTable[A_HI][B_HI]==LO) /*schema at A_HI,B_HI is LO*/
    {
45                  Cval=Cval+hh*0; /* Note hh is effectively being multiplied
                                        by LO value 0 */
    }

    /*repeat this for all combinations of A and B, for example, hh, hm, hl, mh, mm, ml, lh, lm and ll*/
50
```

The resulting score from overlapping values of A and B on the matrix is a

fuzzy value for C that can be defuzzified back into a crisp number. This predictive

value of C is used to search the data set for actual values of C generated during the

16

experiments. To qualify as an actual value of C in the illustrated embodiment, a data

point must fit the predicted value of C in all or nearly all 17 experiments.

At S30, for each triplet, the agreement with the assertion in the rule table can

be calculated based on average square of the residual, r^2, between the calculated C and

5   the observed C for each experiment. Triplets having a low r^2 value fit the assertion

better, and therefore are reported with higher confidence. During initial screening,

only those triplets with $r^2 < 0.0015$ were accepted, corresponding to an average error of

3% or less. This value is well below the error associated with the expression data.

The value of r^2 was set *a priori* to minimize the amount of acceptable subsets, and

10   may be varied according to the stringency requirements of a given screening process.

The following shows how fuzzified data may be applied to a decision matrix

for a particular example, as shown in the pseudocode. The example shows A having a

fuzzy value of 1.0 HIGH, 0 MED and 0 LOW and a B having a fuzzy value of 0.5

HIGH, 0.5 MED and 0 LOW.

15

A: 1.0 HI, 0 MED, 0 LO
B: 0.5 HI, 0.5 MD, 0 LO

Filling out the matrix for all 9 possible values of A and B, i.e. hh, hm, hl, mh, mm, ml, lh, lm and ll, one
20   can see that overlaps at fuzzA[HI]fuzzB[HI] (or hh) and fuzzA[HI]fuzzB[MED] (or hm) count=2
hh=1+0.5
hm=1+0.5
hl=0;
mh=0;
25   mm=0;
....
ll=0;

count=2, thus divide all by 3
30

hh=0.5
hm=0.5
hl=0;
.....
35   ll=0;

thus always has a sum of 1.

For the activator repressor schema, TempTable[A_HI][B_HI]=HI and

TempTable[A_HI][B_MD]=HI.

Thus Cval=0.5+0.5=1 /* Cval = hh +hm */

5　　Meaning that for this activator/repressor couple, we would expect C to be
1 (which means HI expression).
If in truth, Cexperiment=0.98, then we calculate an r^2 of
(1-0.98)^2=0.0004

10

Occasionally, the data set of A and B failed to properly explore the decision

matrix (i.e. A is almost always high, and B is almost always low), thus a second score

called the variance was also assigned to the data set. The variance is defined as the

statistical variance between the total hits in each box on the decision matrix. If the

15　　data set hits are evenly distributed throughout the decision matrix, then the variance

score is low and the resulting predictions are credible. However, if the data set is

poorly distributed, then the variance will be high and the predictions may or may not

be believable due to the lack of combinations of A and B tested. The calculation of

variance for the illustrated example, as referred to by S32 of Fig. 11, is set forth in the

20　　following pseudocode:

```
/*after each experiment, we have values of hh, hm, hl, mh, mm, ml, lh, lm and ll which are put into a
3x3 table, called tablehold[2][2].  For the next experiment, new values of hh, hm, etc. are generated
25      which are added to the current values of tablehold[2][2].  Once all experiments are run (17 in the case
of the yeast data) then we use the 9 values in tablehold[2][2] to find how well distributed the values are.
If all data is in HI,HI then we expect to get a poor distribution (i.e. tablehold[HI][HI] is very large and
all other entries are low), and thus a high variance. */

30      /*calculate mean values */
mean=(tablehold[HI][HI]+tablehold[HI][MED]+ . + tablehold[LO][LO])/9;

/*calculate variance*/
variance=((tablehold[HI][HI]-mean)*(tablehold[HI][HI]-mean)+
35          (tablehold[HI][MED]-mean)*(tablehold[HI][MED]-mean)+
(tablehold[HI][LO]-mean)*(tablehold[HI][LO]-mean)+
.....
(tablehold[LO][LO]-mean)*(tablehold[LO][LO]-mean))/9;
```

40　　As referred to by S34 of Figure 11, a global view of how well the assertion fits

the data may be gleaned from multiplying the r^2 value and the variance to give a

18

general score. Thus, triplets with low r^2 values and low variance have the lowest

score and also should be the most credible statements. Other data sets that are only

low in one parameter may be filtered out because either the data set is biased or the fit

is too poor. For the initial screen, only those A/B pairs with a variance of 1.5 or less

5   were accepted.

Of the 6,321 known proteins in yeast, only 1898 (30%) genes were found

whose expression levels were above the noise threshold and had a maximum value at

least 3 times greater than the minimum value, ensuring that the observed signal

change was statistically significant. Only these qualifying genes were processed by

10   the fuzzy logic algorithm. Using an initial screening cutoff for r^2 and variance, the list

of potential genes was narrowed down to the top 0.007% of all possible triplets,

thereby forming the basis for initial calculations.

In order to evaluate and validate the algorithm, the best scoring triplets were

examined to assess their biological relevance.. One of the best scoring triplets, CYB2-

15   HAP1-CYC7, is shown in Figure 3., wherein the top panel shows a correlation of the

predicted and the observed values for CYC1 (C), while the middle panel shows the

relationship of CYB2 (A) and HAP1 (B), and the bottom panel depicts the

relationship of CYB2 (A) and CYC1 (C).

Figure 3A shows the tight fit of the fuzzy logic prediction for CYC7

20   expression level as compared to the experimental data. The r^2 value for this triplet is

0.013 between the calculated expression data of CYC7 and the observed data (Figure

3A), indicating a very high confidence for the correlation. Moreover, the expression

data of CYB2 (A) and HAP (B) show a fairly wide range of values and are evenly

distributed throughout the decision matrix (Figure 3C). Thus, the variance score is

25   low and the resulting predictions are credible. Of note is the fact that neither CYB2

(A) nor HAP1 (B) could be categorized in the same cluster as CYC7 (C) as shown in Figure 3b. Thus, only by using the fuzzy logic algorithm of the illustrated embodiments of the invention could this triplet be uncovered.

Previous studies show that all three genes in this triplet are involved in yeast

5   respiration process and the predicted relationships between the genes are borne out by experimental results in the literature (Fytlovich, 1993; Lodi, 1991). The transcription factor HAP1 has been shown to repress the nuclear encoding cytochrome gene CYC7 under anaerobic growth, but activate CYC7 under aerobic growth (Prezant, 1987). The fuzzy logic prediction suggests that HAP1 represses CYC7, which in turn

10   accurately predicts that the cells used in this data set were primarily grown under anaerobic conditions.

The algorithm also predicts that CYB2 should activate CYC7, again in agreement with experimental findings. CYB2, L-(+)-lactate cytochrome c oxidoreductase is a soluble protein from the intermembrane of mitochondria. This

15   protein transfers electrons from L-(+)-lactate to cytochrome c and is upstream of cytochrome c on the electron transport chain. Experimental findings indicate that CYB2 interacts preferentially with CYC7 during the electron transfer process (Fytlovich, 1993) and as such should positively regulate the expression of CYC7 as found by the algorithm.

20   In further exploring the relationships revealed by this triplet, all triplets containing either HAP1 or HAP1 regulated genes were selected in an effort to generate an interconnected network describing the control roles of HAP1. The network predicted by the fuzzy logic algorithm is described in Figure 4 and is highly consistent with the experimental data obtained from previous studies. Moreover, the

25   fuzzy logic algorithm permitted functional identification of unidentified genes

20

involved in this process, enabling hypothesis formation for future experimental tests. For example, previous studies show that an unidentified protein X masks the activation domain of HAP1 and allows HAP1 to act as a repressor under anaerobic conditions (Zhang, 1998; Hatch, 1999). Currently, protein X remains uncharacterized.

5   However, fuzzy logic prediction suggests that several gene products, including YDL174C, YGL037C, YLR251W, YLR252W and YNL007C, could be this uncharacterized protein. Functionally, these proteins appear to repress HAP1's ability to activate CYC7, however further experiments are needed to determine the exact protein involved.

10   Both CYT1 and CYC1 are regulated by HAP1 (Schneider, 1991), however the fuzzy logic algorithm only uncovered a relationship for CYT1. On further study it was found that CYT1, like CYC7 discussed above, has a single HAP1 binding site, while the CYC1 promoter has two HAP1 binding sites (Prezant, 1987). A major consequence of the two cooperative sites of CYC1 is that the HAP1 modulation

15   profile should be sigmoidal rather than linear. By contrast, the single promoter site in CYT1 and CYC7 should provide these genes a more linear reduction in activity as HAP1 expression changes. The fuzzy logic algorithm of the claimed invention is preferably limited to detecting linear relationships, although nonlinear proteins controlled by two redundant sites may be detected with less efficiency than linear

20   relationships.

Experimentally, it has been shown that HAP1 regulates ROX1, a protein that encodes a repressor protein for hypoxic genes (Zitomer, 1997, Deckert, 1995). When cells are grown under aerobic conditions, heme accumulates to levels sufficient to induce ROX1 expression and the hypoxic genes are repressed. When cells are limited

25   for oxygen, heme levels fall, ROX1 repressor levels are reduced and hypoxic gene

21

expression is depressed. The relationship between HAP1 and ROX1 was not revealed

by the fuzzy logic because expression of ROX1 is transient and highly unstable. Two

genes, CYT1 and GPD2 were, however, found by the algorithm as the targets for the

positive regulation by ROX1 while most of the hypoxic genes were not identified.

5   This result suggests that the view of hypoxic gene regulation is correct in terms of the

phenomenology but there is a great deal more complexity to ROX1 regulation.

Table 1 lists the most frequent occurring pairs of genes in triplets identified by

the fuzzy logic algorithm, many of which appear to be biologically relevant. In

several cases, both gene products function in the same cellular process. For example,

10   AGP1 and MEP2 are often found together. Functionally AGP1 encodes a broad

substrate range amino acid permease whose expression is subject to nitrogen

repression, while MEP2 is a high-affinity ammonia permease induced by to nitrogen

starvation. Thus, it makes sense that AGP1 and MED2 are found in the same triplet.

Similarly, HAP1 is a transcription factor with a broad spectrum of targets including

15   genes involved in sterol biosynthesis such as FAA1 and ARE2. FAA1 is long chain

fatty acyl:CoA aynthetase in lipid metabolism and protein N-myristolation. ARE2 is

sterol-ester synthetase in ergosterol esterification. In general, HAP1 is known as a

repressor, thus the fact that this algorithm identifies HAP1 as repressing FAA1 and

ARE2 is consistent with known biological data.

20   YGP1 and CBF2 represent a very interesting relationship predicted by the

algorithm. YGP1 is a highly glycosylated secreted protein involved in cellular

adaptations prior to stationary phase. The gene is expressed at a basal level during

logarithmic growth and induced up to 50-fold above basal level when cells enter

stationary phase. Conversely, CBF2 is a chromosome centromere binding protein in

25   the multisubunit protein complex and involved in cellular replication and cell

22

division. CBF2 expression is expected to be high during logarithmic growth, but low during stationary phase. This reciprocal relationship between YGP1 and CBF2 is well represented in the predicted results as the two proteins were found in either B or C positions depending upon the protein as the activator in the triplet. When CDC46, a

5   protein enriched in non-dividing cells, is the activator, YGP1 is found as the target in the triplet. When proteins promoting cell cycle progression such as CDC45, RPL14A, ZDS2, NHP6A and IPL1 are activators, CBF2 acts as the target.

In addition, pairs of genes are also predicted by the fuzzy logic algorithm wherein one or both of the genes are uncharacterized. By analogy to the examples

10   shown above, it may be possible to infer the cellular function of these unknown proteins by examining what known proteins are found to associate with the set. This ability to bootstrap functional information out of the expression data should be particularly useful in analyzing human data, where a much larger percentage of proteins are uncharacterized.

15   Since the fuzzy logic algorithm of the illustrated embodiment was used to search for activator-repressor-target triplets, a disproportionately large number of triplets were expected to include transcription factors. After an initial screening, the results revealed that transcription factors were found 30% more frequently than would be expected from a random distribution. When only looking at the 100 best scoring

20   triplets, transcription factors were found 90% more frequently than at random.

In general, the findings of the algorithm agree well with experimental results from the literature. Such findings are rational, given that the algorithm searches for relationships that fit logical scientific understanding of how an activator, repressor, and target should interact. The fuzzy logic algorithm embodiments illustrated herein

25   only searched for triplets of activator, repressor, and target genes. The choice of the

activator-repressor model was based on simplicity in order to demonstrate that this

technology is capable of yielding biologically meaningful results. The technique,

however, is not limited to triplet subsets, and can be applied to different size subsets

of data, corresponding to other relationships and more complicated systems.

5   Examples include, *inter alia*, other classes of relationships such as co-activators and

co-repressors, or more complicated systems that involve genes whose transcription is

regulated by any number of transcription factors. It is contemplated that the

technology may be useful in describing complete general networks of gene

interactions.

10   Using fuzzy logic to analyze expression data does have some limitations. To a

first approximation, the interaction between multiple proteins is essentially linear, thus

the algorithm searches for linear behavior. However, in the event of multiple

redundant promoter binding sites (such as the two HAP1 binding sites on the CYC1

gene), this linear approximation is not accurate, causing the algorithm to overlook

15   these biologically relevant connections. Such a situation could be remedied by

including a more sophisticated "fuzzification" step to include nonlinear effects. While

useful, this added complexity may only correct for a few missed connections while

edging out many of the more common near linear relationships. Still, gross nonlinear

relationships between biological moieties may be assessed using fuzzy logic.

20   In practice, the fuzzy logic algorithm found a disproportionately large number

of transcription factors in the roles of activators and repressors, although not all of the

activators and repressors found were transcription factors. Two possible reasons for

this discrepancy are 1) transcription factors are expressed at low levels and thus are

difficult to detect; and 2) enzymes can indirectly regulate transcription. Transcription

25   factors are generally present only at a very low concentration, thus changes in

transcription factor expression levels can be difficult to detect using current

expression profiling techniques. Presumably, if expression profiling technology were

to become more sensitive, the fuzzy logic algorithm would detect an even greater bias

of transcription factors in the activator and repressor roles. However, in many cases

5   the expression level of a particular protein is not governed by the expression of a

transcription factor, but instead by the concentration of some intracellular compound,

such as Ca2+ concentration or cAMP levels, which in turn are controlled by enzymes

inside the cell. In these cases, changes in the expression level of the enzyme have a

"transcription-factor-like" effect and would be detected by the algorithm as an

10   activator or repressor. From an drug design point of view, these "transcription-factor-

like" enzymes are possibly more interesting than true transcription factors, because it

is generally easier to change the activity of an enzyme in the cytosol with a drug than

to block a true transcription factor in the nucleus.

Although the validation of this algorithm was performed using GeneChip®

15   data, the fuzzy logic algorithm should work equally well with other expression

profiling techniques such as Sequential Analysis of Gene Expression (SAGE). SAGE

has the advantage that it can detect completely unknown proteins, while GeneChip®

technologies require that at least the sequence of a protein's mRNA be known. This

ability to detect unknown proteins would be particularly well suited to the functional

20   characterization that the fuzzy logic algorithm makes possible.

An additional advantage to the fuzzy logic algorithm is that data can come

from any source within an organism (tissue, cell type, treatment, or physiological

state) and the output actually will be improved by deeper and more diverse data set.

The reason for this improvement is that the algorithm needs to observe changes in the

25   expression level of a protein relative to changes in other expression levels. Each new

data set provides a different set of expression levels that can be tested to see if they fit the proposed regulatory model. In our studies, many data sets were eliminated solely because they did not sufficiently explore the combinations of expression levels (too high a sigma value), making their predictions impossible to believe. By including

5   data sets from cells in different states, the algorithm gains more information about the details of the regulatory network.

One application of this algorithm is to independently validate or discover drug targets. Traditional techniques for drug target discovery require a detailed understanding of the biology underlying the disease, which can be slow and difficult

10   to obtain. In contrast, expression profiling is a rapid high-throughput process, providing a large amount of information about the cell in a form that could be easily processed on a computer. By using a fuzzy logic approach to analyzing expression profile data, it is possible to confirm the mechanism of a known target. Moreover, because the fuzzy logic algorithm does not require biological information about the

15   gene, genes encoding proteins with unknown functions can be included just as easily as genes encoding proteins with known functions. This ability to identify functional clues for uncharacterized genes is a great advantage in drug target discovery because potential drug targets then can be followed up with the detailed biology.

The invention may be implemented by hardware or a combination of hardware

20   and software. The software may be recorded on a medium for reading into a computer memory and executing. The medium may be, but is not limited to, for example, one or more of a floppy disk, a CD ROM, a writable CD, a Read-Only-Memory (ROM), and an Electrically Alterable Programmable Read Only Memory (EAPROM).

While the invention has been described by way of example embodiments, it is

25   understood that the words which have been used herein are words of description, rather

26

than words of limitation. Changes may be made, within the purview of the appended claims, without departing from the scope and spirit of the invention in its broader aspects. Although the invention has been described herein with reference to particular means, materials, and embodiments, it is understood that the invention is not limited to

5   the particulars disclosed. The invention extends to all equivalent structures, means, and uses which are within the scope of the appended claims.

REFERENCES

1. **Cho, R. J., M.J. Campbell, E. A. Winzeler, L. Steinmetz, A. Conway, L. Wodicka, T. G. Wolfsberg, A. E. Gabrielian, D. Landsman, D. J. Lockhart,**

10   **and R. W. Davis.** A Genome-Wide Transcriptional Analysis of the Mitotic Cell Cycle. *Mol. Cell* 2:65-73, 1998.

2. **Cox, E.** Fuzzy Fundamentals. *IEEE Spectrum*, 58-61, 1992

3. **Deckert, J., R. Perini, B. Balasubramanian, and R. S. Zitomer.** Multiple Elements and Auto-Repression Regulate Rox1, a Repressor of Hypoxic Genes in

15   *Saccharomyces cerevisiae. Genetics Soc. Of America* 139:1149-1158, 1995.

4. **Fytlovich, S., M. Gervais, C. Agrimonti, and B. Guiard.** Evidence for an interaction between the CYP1(HAP1) activator and a cellular factor during heme-dependent transcriptional regulation in the yeast *Saccharomyces cerevisiae. EMBO J.* 12:1209-1218, 1993.

20   5. **Hach, A., T. Hon, L. Zhang.** A New Class of Repression Modules Is Critical for Heme Regulation of the Yeast Transcriptional Activator Hap1. *Mol. Cell. Biol.* 19:4324-4333, 1999

6. **Lodi, T., and B. Guiard.** Complex Transcriptional Regulation of the *Saccharomyces cerevisiae* CYB2 Gene Encoding Cytochrome b_2: CYP1(HAP1)

Activator Binds to the CYB2 Upstream Activation Site UAS1-B2. *Mol. Cell. Biol.* 11:3762-3772, 1991.

7. **Prezant, T., K. Pfeifer, and L. Guarente.** Organization of the Regulatory Region of the Yeast CYC7 Gene: Multiple Factors Are Involved in Regulation. *Mol. Cell. Biol.* 7:3252-3259, 1987.

8. **Schneider, J. C., and L. Guarente.** Regulation of the Yeast CYT1 Gene Encoding Cytochrome c_1 by HAP1 and HAP2/3/4. *Mol. Cell. Biol.* 11:4934-4942, 1991.

9. **Tavazoie, S., J. D. Hughes, M. J. Campbell, R. J. Cho, and G. M. Church.** Systematic Determination of Genetic Network Architecture. Nature Genetics 22:281-285, 1999.

10. **Yakhini, Z., and Ben-Dor, A.** Clustering Gene Expression Patterns. Technical Report HPL-98-190, HP-labs Israel, 1998.

11. **Zadeh, L. A.** Fuzzy Logic and its Application to Approximate Reasoning. *Information Processing,* 74:591-594, 1974.

12. **Zhang, L., A. Hach, and C. Wang.** Molecular Mechanism Governing Heme Signaling in Yeast: a Higher-Order Complex Mediates Heme Regulation of the Transcriptional Activator HAP1. *Mol. Cell. Biol.* 18:3819-3828, 1998.

13. **Zitomer, R. S., M. P. Limbach, A. M. Rodriguez-Torres, B. Balasubramanian, J. Deckert, and P. M. Snow.** Approaches to the Study of Rox1 Repression of the Hypoxic Genes in *the Yeast Saccharomyces cerevisiae. Methods in Enzymology* 11:279-288, 1997.

Table 1: Frequent pairs of genes identified by the fuzzy logic algorithm

Name	Position	Description	# repeats
AGP1	A	Broad substrate range permease which transports asparagine and glutamine	
MEP2	C	Ammonia transport protein	79
YGP1	B or C	involved in cellular adaptations prior to stationary phase	
CBF2	B or C	component (Cbf3a) of the multisubunit 'Cbf3' kinetochore protein complex	71
PEP5	C	peripheral vaculor membrane protein; putative Zn-finger protein	
MEP2	A	Ammonia transport protein	66
PEP5	C	peripheral vaculor membrane protein; putative Zn-finger protein	
ARG4	A	argininosuccinate lyase	63
CPS1	B	carboxypeptidase yscS	
HES1	C	Protein implicated in ergosterol biosynthesis	57
GAP1	B	general amino acid permease	
SPO13	C	a negative regulator of M-phase during meiosis	52
HAP1	B	positive regulator of cytochrome C genes CYC1 and CYC7	
FAA1	C	long chain fatty acyl:CoA synthetase	44
HAP1	B	positive regulator of cytochrome C genes CYC1 and CYC7	
ARE2	C	Acyl-CoA cholesterol acyltransferase (sterol-ester synthetase)	39
HAP1	B	positive regulator of cytochrome C genes CYC1 and CYC7	
GAP1	C	general amino acid permease	38

29